Multiclass scheduling algorithms for the DAVID metro network

Multiclass Scheduling Algorithms for the DAVID Metro Network

Andrea Bianco, *Member, IEEE*, Davide Careglio, Jorge M. Finochietto, *Student Member, IEEE*, Giulio Galante, *Member, IEEE*, Emilio Leonardi, *Member, IEEE*, Fabio Neri, *Member, IEEE*, Josep Solé-Pareta, *Member, IEEE*, and Salvatore Spadaro

***Abstract*—The data and voice integration over dense wavelength-division-multiplexing (DAVID) project proposes a metro network architecture based on several wavelength-division-multiplexing (WDM) rings interconnected via a bufferless optical switch called Hub. The Hub provides a programmable interconnection among rings on the basis of the outcome of a scheduling algorithm. Nodes connected to rings groom traffic from Internet protocol routers and Ethernet switches and share ring resources. In this paper, we address the problem of designing efficient centralized scheduling algorithms for supporting multiclass traffic services in the DAVID metro network. Two traffic classes are considered: a best-effort class, and a high-priority class with bandwidth guarantees. We define the multiclass scheduling problem at the Hub considering two different node architectures: a simpler one that relies on a complete separation between transmission and reception resources (i.e., WDM channels) and a more complex one in which nodes fully share transmission and reception channels using an erasure stage to drop received packets, thereby allowing wavelength reuse. We propose both optimum and heuristic solutions, and evaluate their performance by simulation, showing that heuristic solutions exhibit a behavior very close to the optimum solution.**

***Index Terms*—Data and voice integration over dense wavelength-division multiplexing (DAVID), metropolitan area network, multiclass scheduling, optical ring, wavelength-division multiplexing (WDM).**

I. Introduction

THE DATA and voice integration over dense wavelength-division multiplexing (DAVID) project, funded by the Information Society Technologies (IST) program of the European Commission, aimed at providing data and voice integration over an optical dense wavelength-division-multiplexing (WDM) packet-switched network, by developing innovative concepts and technologies for optical networks [1]. As shown in Fig. 1, DAVID encompasses both metropolitan and long haul geographical scales and is based on a hierarchical architecture consisting of several optical metro networks interconnected through an optical backbone.

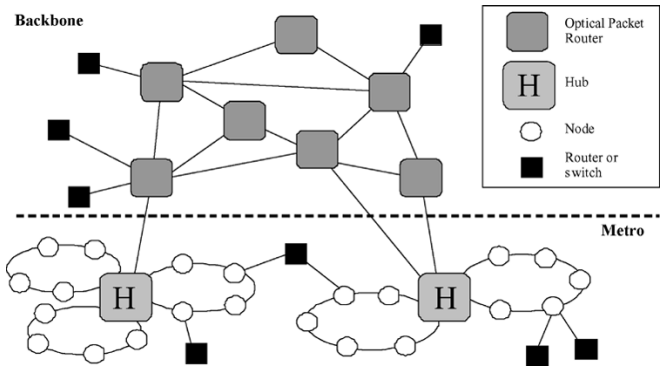


Fig. 1. Overview of the DAVID network.

In this paper, we focus on the DAVID *metro* network, which consists of several unidirectional WDM rings interconnected in a star topology by an optical bufferless switch called *Hub*. The Hub connects the rings in a metro network to at least one *optical packet router* in the backbone. Nodes are connected to rings and groom traffic from Internet protocol (IP) routers and Ethernet switches. Nodes belonging to the same ring share a fixed number of wavelengths. The information on the status and availability of network resources is transported on a dedicated additional wavelength on each ring named *control channel*.

Two different node architectures were studied within DAVID: a simpler one that relies on a complete separation between transmission and reception wavelengths, and a more complex one in which nodes fully share transmission and reception channels using an erasure stage to drop received packets, so that wavelengths can be reused.

No packet buffering is available at the Hub; therefore, buffering must be done electronically at nodes. The Hub behaves as a switch between rings. Ring interconnections are dynamically modified at the Hub following a scheduling algorithm, whose aim is to provide node pairs with a portion of the network capacity close to short-term bandwidth requirements. The scheduling is based on a traffic request matrix created at the Hub on the basis of explicit bandwidth requests issued by nodes, as an alternative, traffic matrices can be estimated with measurements at the Hub, as discussed in [2].

Since it is reasonable to think that in future metro networks multimedia and interactive applications will take an important

Manuscript received August 14, 2003; revised March 15, 2004. This work was supported in part by the European Commission under the IST DAVID Project IST 1999-11742 and in part by the Spanish Ministry of Science and Technology MCYT) under Contract FEDER-TIC2002-04344-C02-02.

A. Bianco, J. M. Finochietto, E. Leonardi, and F. Neri are with Dipartimento di Elettronica, Politecnico di Torino, 10129 Torino, Italy (e-mail: bianco@polito.it; finochietto@polito.it; leonardi@polito.it; neri@polito.it).

D. Careglio, J. Solé-Pareta, and S. Spadaro are with the Advanced Broadband Communications Centre (CCABA), Universitat Politècnica de Catalunya, Barcelona 08034, Spain (e-mail: careglio@ac.upc.es; pareta@ac.upc.es; spadaro@tsc.upc.es).

G. Galante was with Dipartimento di Elettronica, Politecnico di Torino, 10129 Torino, Italy. He is now with Istituto Superiore Mario Boella, 10138 Torino, Italy (e-mail: galante@ismb.it).

Digital Object Identifier 10.1109/JSAC.2004.830502

share of the bandwidth, techniques to provide quality of service (QoS) must be designed. The simplest way to do that is by introducing at least two traffic classes having different priorities [3]. A drawback of such strategy is that if high-priority (HP) traffic is not controlled by any form of call admission control, traffic fluctuations can cause two different undesirable situations: 1) HP traffic with strict priority over low-priority traffic can prevent the transmission of the latter and 2) it may not be possible to guarantee neither delay nor bandwidth bounds even to HP traffic. To avoid this situation in public networks, centralized bandwidth management functions (i.e., traffic engineering) are required. Network operators need to have the possibility to control the amount of HP traffic injected in their network.

In this context, we address the problem of designing a centralized scheduling algorithm at the Hub capable of supporting multiclass traffic. In particular, we consider two service categories, namely, a conventional low-priority best-effort service and an on-demand bandwidth-guaranteed service to support HP traffic for applications such as Internet telephony, video broadcasting, and videoconferences.

The remainder of the paper is organized as follows. In Section II, we provide a more detailed description of the DAVID metro network, focusing on the Hub operation and on the two node architectures dubbed *frequency-decoupling* and *full-sharing*. In Section III, we discuss the multiclass scheduling problem. Then, in Section IV, we provide an integer linear programming (ILP) formulation of the multiclass scheduling problem for the frequency-decoupling architecture, and present an optimum scheduling algorithm with polynomial computational complexity. In Section V, we present a heuristic solution for the frequency-decoupling architecture, with the aim of decreasing the complexity of the optimum solution. In Section VI, we describe a heuristic algorithm for the full-shared node architecture. Finally, in Section VII, we present simulation results to assess the performance of the proposed solutions. Section VIII concludes the paper.

II. DAVID Metro Network Architecture

In this section, we describe the network architecture studied in this paper. We do not tackle issues related to the metro network feasibility in this paper, nor do we discuss the components that should be used or the physical limitations that should be taken into account when dimensioning the network. All these issues have been deeply analyzed in the DAVID project and have been discussed in [1].

A. General Overview

As stated above, the DAVID metro network consists of several WDM rings interconnected through a Hub. The number of rings connected to the Hub is denoted by R while, for simplicity, the number of nodes on each ring is assumed to be equal and denoted by n, so that the total number of nodes is $N = Rn$.

Nodes of the metro network share ring resources using a statistical time/wavelength/space division multiple access scheme. Indeed, time is divided into slots lasting 500 ns–1 μs [time-division multiple access (TDMA)]. Several slots are simultaneously transmitted on different wavelengths on the same ring [wavelength-division multiple access (WDMA)], and rings are disjoint in space [space-division multiple access (SDMA)]. Although the number of wavelengths on a ring may in general vary from ring to ring, here, for simplicity, we assume that each ring conveys the same number of wavelengths.

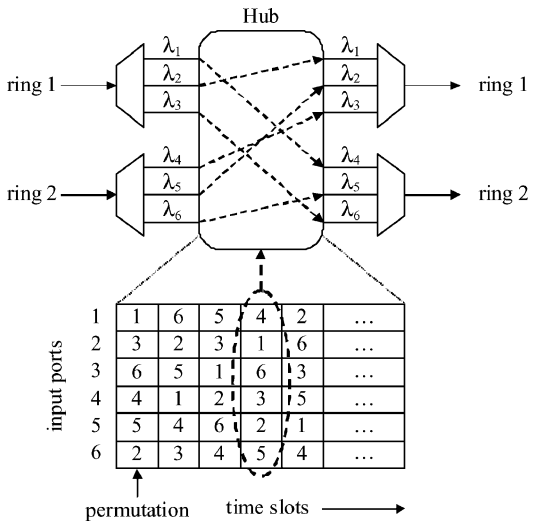


Fig. 2. Example of a sequence of wavelength-to-wavelength permutations.

The network is synchronous and time slots are aligned on all wavelengths of the same ring; thus, a *multislot* (i.e., a set of slots, one per wavelength) is available at each node in any time slot. For simplicity, the propagation delay on each ring is assumed to be an integer multiple of the slot duration. Each multislot includes one *control slot* and several *data slots*. Control slots contain information for regulating node access and ring interconnections, whereas data slots transport user data.

B. Hub Architecture and Operation

The Hub is a bufferless optical switch that connects rings. In general, the Hub can set up *wavelength-to-wavelength permutations* among rings, held for one time slot. Alternatively, Hub permutations can be *ring-to-ring*, so that all slots on the different wavelengths coming from a given ring are switched to the same output ring. Fig. 2 shows a metro network comprising two rings conveying three wavelengths each, where the Hub operates wavelength-to-wavelength permutations. Since the Hub is all optical, it only includes a space switching stage, a wavelength conversion stage, and a WDM synchronization module to align slots; 3R regeneration may be added if required by physical layer constraints. The details of the technological implementation of the Hub architecture are described in [1].

The Hub collects bandwidth requests issued by nodes, builds a node-to-node traffic request matrix, and decomposes it in either a sequence of wavelength-to-wavelength permutations or in a sequence of ring-to-ring permutations using a scheduling algorithm [4]–[8].

In this paper, we assume that permutations are organized in a fixed-length frame, and are repeated cyclically until a new traffic matrix is computed from user reservations. Note that the time-scale upon which bandwidth requests are collected from users may be much longer than the duration of a single frame, decreasing the computational load on the Hub. The problem of

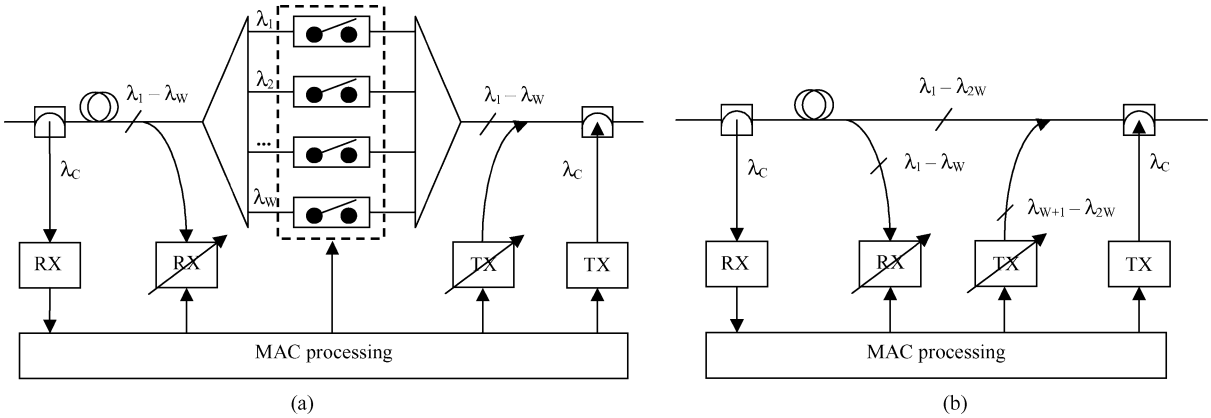


Fig. 3. Architecture of (a) an FS node with erasure capability and (b) an FD node with transmission and reception decoupling.

scheduling multiclass traffic in the DAVID metro network is thoroughly discussed in Section III.

C. Node Architecture and Operation

Each node comprises an electronic and an optical part. Packets coming from routers and switches are stored in the electronic part of the nodes. To avoid the head-of-line blocking phenomenon [9], packets are organized per traffic class and stored in a per-destination-node queue architecture, similar to the virtual output queue architecture adopted in input-queued packet switches [10].

The optical part of each node consists of a fixed transceiver to read and write control slots and a tunable transceiver to access data wavelengths. A fiber delay line must be added at each node along the optical path of data wavelengths, so as to allow enough time for terminating, electronically processing, and regenerating the control channel.

Aiming at a balance between optical and electronic complexities, the transmission and reception bandwidth at each node is equal to one wavelength, i.e., a node cannot simultaneously transmit nor receive on more than one wavelength. However, since tuning latencies are negligible compared with the slot duration, transceivers can switch to different wavelengths on a slot-by-slot basis.

A scheduling algorithm arbitrates the allocation of network resources and nodes' access to slots. It is typically run in a centralized fashion at the Hub, although some decisions can be decentralized at network nodes, when the Hub implements ring-to-ring permutations. Nodes must have access opportunities in proportion to the requested traffic matrix while avoiding *collisions* (i.e., transmissions in busy slots) and *contentions* (i.e., sending in different slots of the same multislot more than one packet addressed to the same node).

As mentioned earlier, information on the status of each slot in a multislot is available in the control slot. Therefore, a node that has packets to send must monitor the control slot to implement access decisions, and, at the same time, look for any instance of its address for possible receptions. Note that only the in-transit information on the control channel is converted to the electrical domain to be processed at nodes, while the rest of the information remains in the optical domain along the entire source-destination path.

Two node architectures were considered in the DAVID project. In the first one, shown in Fig. 3(a), nodes have a selective erasure capability, which allows packet removal at the destination, hence, wavelength reuse (i.e., packets only circulate along ring spans between source and destination nodes). We call this solution *full-sharing* (FS), since nodes use the same set of W wavelengths for both transmission and reception. One more wavelength is needed for the control channel, bringing the number of wavelengths on each ring to $W + 1$. Packet erasure is costly in terms of optical components and adds physical layer impairments to the data path, limiting the total number of nodes on the ring [11].

The second node architecture, depicted in Fig. 3(b), is instead lacking the packet drop stage, so that nodes keep all packets on the ring, simply copying packets addressed to them. Unfortunately, this approach may leave little or no room for new packets to be transmitted on the ring, although this may be easily overcome by completely separating transmission and reception channels. We call this solution *frequency-decoupling* (FD) because nodes access the ring using separate sets of wavelengths: W for transmission, W for reception, and one for network control; thus, the total number of wavelengths conveyed on each ring becomes $2W + 1$. Data are sent by nodes on transmission wavelengths and switched from transmission wavelengths to reception wavelengths at the Hub, which must provide wavelength conversion. Packets are received by nodes from reception wavelengths and are dropped when they reach the Hub, which, in turn, generates empty transmission channels for downstream nodes.

Note that, while FS can benefit from wavelength reuse, FD can lead to bandwidth waste on the network due to the fixed partitioning of resources between transmission and reception channels. However, although FS usually achieves better network performance and, in general, needs fewer wavelengths, FD requires less optical components, and no active components are inserted along the optical data path for packet erasure.

D. Network Operation

Fig. 4 illustrates the operation of a network equipped with FS nodes when the Hub performs wavelength-to-wavelength permutations. For simplicity, we consider a network with $R = 4$ rings, $W = 3$ wavelengths per ring, and

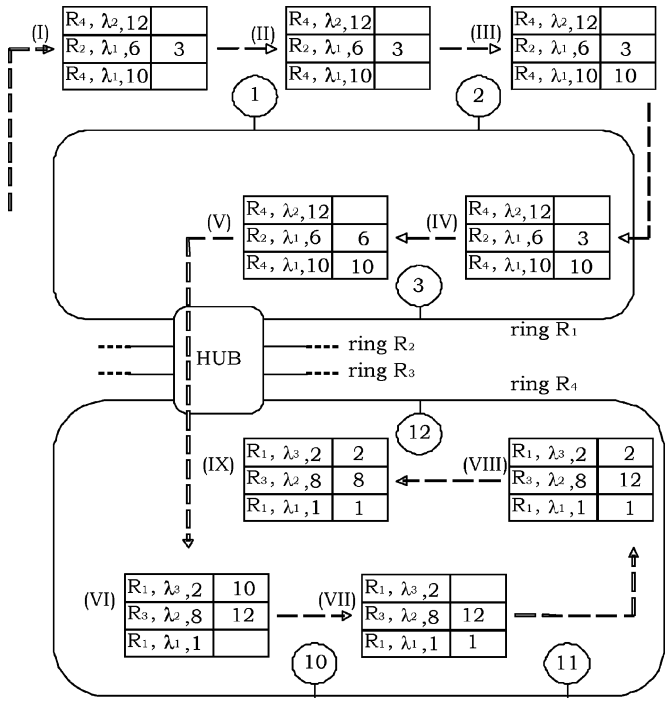


Fig. 4. Example of slot forwarding in the metro network.

$n = 3$ nodes per ring. Control slots are on the left-hand side of multislots, whereas data slots are on the right-hand side. Data slots are reserved for packets addressed to nodes identified in the triple (ring, wavelength, node) inserted by the scheduling algorithm running at the Hub in the corresponding control slot. Numbers in data slots denote the destination node to which the packets they convey are addressed. We describe the slot forwarding process in successive steps showing the path followed by a multislot as it travels from ring R_1 to ring R_4.

I) Control slots indicate that the first data slot is reserved for sending data to node 12 on ring R_4 using wavelength λ_2, the second data slot is reserved for a packet to be delivered on λ_1 to node 6 on R_2, and the third slot is booked for transporting on λ_1 a packet addressed to node 10 on R_4. The second data slot is carrying a packet headed to node 3.

II) Node 1 leaves the multislot untouched.

III) Node 2 has a packet for node 10 on ring R_4 and inserts it in the corresponding data slot.

IV) Node 3 drops the packet transported in the second data slot and reuses it to send a packet to node 6 on ring R_2.

V) The multislot reaches the Hub.

VI) The Hub switches data slots to their intended output ring/wavelength as indicated by the triple contained in the relevant control slot. Therefore, the third slot containing a packet headed to node 10 is switched to wavelength λ_1 (first slot) of ring R_4, and a packet coming from another ring and with destination node 12 is inserted in the second slot. Finally, the scheduling algorithm inserts new destination triples in the control slots.

VII) Node 10 drops the packet sent from node 2 and transmits a packet to node 1 using the third data slot.

VIII) Node 11 transmits a packet to node 2 on ring R_1 using the first data slot.

IX) Node 12 drops the packet in the second slot, and transmits a new packet to node 8 on ring R_3.

III. Multiclass Scheduling Problem

We consider a HP traffic class with guaranteed bandwidth, and an elastic *best-effort* (BE) traffic class. The Hub collects requests issued by nodes in two matrices: $\mathbf{H}$ for HP traffic and $\mathbf{B}$ for BE traffic. The multiclass scheduling algorithm must address the following issues.

- *Priority of HP traffic over BE traffic*. HP requests must be satisfied before serving BE traffic.
- *Persistent allocation of HP connections*. The allocation of new HP and BE traffic must not affect currently established HP connections.
- *Atomic allocation of HP requests*. HP requests are accepted only when they can be fully satisfied; otherwise, they must be refused. Atomic allocation typically makes sense when requests correspond to single real-time user connections, whereas it does not apply to elastic BE traffic, nor to sources that multiplex several data flows. In this paper, we enforce atomic allocation of HP requests only in the ILP formulation of Section IV-A, which makes the scheduling problem harder to solve.

The architecture of the DAVID metro network introduces two additional constraints.

- *No contentions*. Since each node is equipped with only one tunable data transceiver, it can transmit and receive at most one packet in each multislot. In case of wavelength-to-wavelength permutations, the scheduling algorithm at the Hub must avoid contentions among packets switched by the Hub to their intended destination ring. On the contrary, if the Hub performs only ring-to-ring permutations, contention resolution can be performed in a distributed way at each node.
- *Slot/wavelength reuse*. Slots (and therefore wavelengths) can be reused only with the FS node architecture and the Hub must take into account nodes' positions along the rings, while computing the scheduling. The scheduling problem can be made easier by separating transmissions and receptions either in *frequency* or in *time*. The first approach leads to the FD node architecture, whereas in the second approach, dubbed *time-decoupling*, the same set of wavelengths is shared in the time domain and used alternatively for transmissions and receptions.

The scheduling is based on a fixed-size frame of length F slots, which is considered the most suited solution for a system with guaranteed bandwidth allocation. Indeed, with a fixed-size frame, a reservation issued in terms of bit rate can easily be translated into an equivalent number of slots per frame. Both request matrices $\mathbf{H}$ and $\mathbf{B}$ store the number of slots that must be transmitted from a node to any other node within a frame of F slots. The frame length F must be chosen trading the allocation granularity (asking for longer frames) for the access delay

and scheduling complexity (asking for shorter frames). The selection of optimal values for F is outside the scope of this paper, but we typically envision several thousand slots in the frame.

Scheduling can aim at different levels of performance guarantees. In general, an allocation of slots in the frame should provide average rates to node pairs in accordance with the traffic matrix. The scheme proposed in [2] for BE traffic allocates ring-to-ring rates at the Hub, and access decisions are decentralized at nodes, which, however, do not have guaranteed access. Advantages of that approach are the very small amount of information in the control channel and the good scalability properties.

However, if higher node-to-node guarantees are required, as it is the case for HP traffic, the scheduler must allocate slots to single node-to-node requests. The resource allocation problem becomes mostly centralized, and the amount of information in the control channel increases. It must be noted that a centralized scheduling is often desired by network operators, who may want to apply different control policies for different nodes (e.g., introducing an admission control policy for HP traffic requests).

In the following sections, we describe three solutions to different versions of the multiclass scheduling problem, trying to highlight strengths and weaknesses of each. More precisely, we devise the following.

- Two algorithms for the FD architecture. Section IV introduces an ILP formulation of the multiclass scheduling problem with atomic allocation of HP requests and shows that it is NP-hard, then an optimum polynomial algorithm for the scheduling problem with partial allocation of HP requests is described. Section V introduces a faster greedy algorithm for the same problem that uses wavelength-to-wavelength permutations for HP connections, and ring-to-ring permutations for BE traffic.
- One algorithm for the FS architecture. Section VI describes a greedy algorithm that uses wavelength-to-wavelength permutations for both HP and BE traffic.

IV. Optimum Scheduling for the FD Architecture

A. ILP Formulation

The problem of scheduling multiclass traffic in the DAVID metro network when nodes are in the FD configuration and HP requests must be atomically allocated can be stated as follows.

GIVEN the HP request matrix $\mathbf{H}$, and the BE request matrix $\mathbf{B}$, where $\mathbf{H}_{ij}$ and $\mathbf{B}_{ij}$ are the number of slots to be transmitted form node i to node j,

FIND a slot allocation within a frame of length F,
SUCH THAT:

1) the number of fully allocated HP connections is maximized,
2) the number of slots allocated to BE traffic is maximized.

The problem can be formulated in terms of ILP using the following notation.

l Index used to address slots inside the frame.

$r(i)$ Nodal location function which, for each node i, returns the ring $k = r(i)$ on which node i is located. In the same way, $r^{-1}(k)$ is the set of nodes connected to ring k.

Let h_{ij}^l, b_{ij}^l, and s_{ij} be binary variables defined, as shown at the bottom of the page.

Then, variables h_{ij}^l, b_{ij}^l and s_{ij} must satisfy the following constraints:

$$\sum_j \left(h_{ij}^l + b_{ij}^l\right) \le 1, \quad \forall i, l \tag{1}$$

$$\sum_i \left(h_{ij}^l + b_{ij}^l\right) \le 1, \quad \forall j, l \tag{2}$$

$$\sum_{i \in r^{-1}(k)} \sum_j \left(h_{ij}^l + b_{ij}^l\right) \le W, \quad \forall k, l \tag{3}$$

$$\sum_{j \in r^{-1}(k)} \sum_i \left(h_{ij}^l + b_{ij}^l\right) \le W, \quad \forall k, l \tag{4}$$

$$\sum_l h_{ij}^l = \mathbf{H}_{ij} s_{ij}, \quad \forall i, j \tag{5}$$

$$\sum_l b_{ij}^l \le \mathbf{B}_{ij}, \quad \forall i, j. \tag{6}$$

Constraints (1) and (2) enforce that each node in the network can transmit and receive at most one packet in each time slot. Constraints (3) and (4) make sure that the number of packets transmitted or received by nodes belonging to the same ring in any multislot does not exceed the number of available wavelengths on the ring. Equation (5) enforces atomic allocation of HP traffic. Finally, inequality (6) states that BE requests may be partially satisfied, being considered elastic traffic.

The optimum scheduling is defined as the set of h_{ij}^l and b_{ij}^l that maximizes the number of established HP connections given by

$$\sum_i \sum_j s_{ij} + \varepsilon \sum_i \sum_j \sum_l b_{ij}^l$$

where ε is a positive constant smaller than $1/(FN^2)$. With this choice for ε, the number of slots assigned to BE traffic is maximized only after that the number of HP connections allocated

---

$$h_{ij}^l = \begin{cases} 1, & \text{if node } i \text{ is transmitting an HP packet to node } j \text{ in slot } l \\ 0, & \text{otherwise} \end{cases}$$

$$b_{ij}^l = \begin{cases} 1, & \text{if node } i \text{ is transmitting a BE packet to node } j \text{ in slot } l \\ 0, & \text{otherwise} \end{cases}$$

$$s_{ij} = \begin{cases} 1, & \text{if HP request } \mathbf{H}_{ij} \text{ has been satisfied} \\ 0, & \text{otherwise} \end{cases}$$

is maximum. Note, however, that it would also be possible to maximize the overall network throughput defined as follows:

$$\sum_i \sum_j \mathbf{H}_{ij} s_{ij} + \varepsilon \sum_i \sum_j \sum_l b_{ij}^l.$$

Unfortunately, this formulation does not guarantee the persistency of HP connections. However, such a constraint can be easily enforced by dividing $\mathbf{H}$ into two contributions: $\mathbf{H}^{\text{cur}}$ and $\mathbf{\Delta}$. Element $\mathbf{H}_{ij}^{\text{cur}}$ contains the number of slots allocated to connections currently set up between node i and node j, whereas $\mathbf{\Delta}_{ij}$ is the number of slots to be allocated to new connections to be established between node i and node j. In the model, we need two sets of binary state variables: s_{ij}^{cur} and s_{ij}^{Δ} to distinguish, respectively, between currently established connections and new connections. With this notation, the atomicity constraint (5) becomes

$$\sum_l h_{ij}^l = \mathbf{H}_{ij}^{\text{cur}} s_{ij}^{\text{cur}} + \mathbf{\Delta}_{ij} s_{ij}^{\Delta}, \quad \forall i, j$$

and the objective function can be rewritten as follows:

$$\max \left(\sum_i \sum_j s_{ij}^{\text{cur}} + \varepsilon \sum_i \sum_j s_{ij}^{\Delta} + \varepsilon^2 \sum_i \sum_j \sum_l b_{ij}^l \right).$$

As before, ε is a positive constant smaller than $1/(FN^2)$. In this way, the persistency of currently allocated HP connections is guaranteed, and the number of slots allocated to BE traffic is maximized only after that the number of new HP connections allocated is maximum.

The multiclass scheduling problem for the FD node architecture with atomic HP traffic allocation (5) is NP-hard because it is a generalization of the well-known knapsack optimization problem, which is NP-hard [12]. In fact, HP traffic requests can be considered as a set of objects of different sizes that must be fit in a knapsack of capacity F. When relinquishing the atomicity constraint, the multiclass scheduling problem can be solved in polynomial time, as described in the next subsection.

B. Scheduling Algorithm With No Atomicity Constraint

The multiclass scheduling problem with partial allocation of HP connections can be divided into two polynomial subproblems as described in [13]: the *F-matching* and the *time slot assignment* (TSA).

The F-matching problem consists of finding an admissible subset of HP and BE requests that can be scheduled in a frame of length F, eventually dropping some of the original requests.

Scheduling an admissible request matrix using wavelength-to-wavelength permutations is equivalent to finding a TSA in a *hierarchical switching system* (HSS).

According to the representation of a HSS proposed in [14] (see Fig. 5), a HSS comprises a $WR \times WR$ nonblocking switch, which interconnects several input multiplexers and output demultiplexers. The nonblocking switch models the Hub, whereas each group of W_i inputs and W_i outputs represents the W_i wavelength conveyed by ring i. Multiplexer i models ring i,

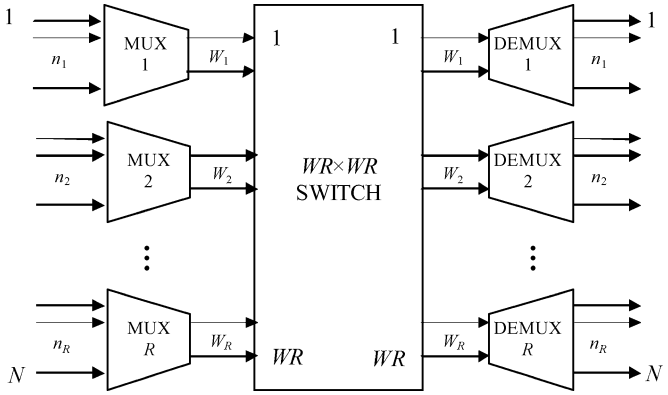


Fig. 5. Hierarchical switching system.

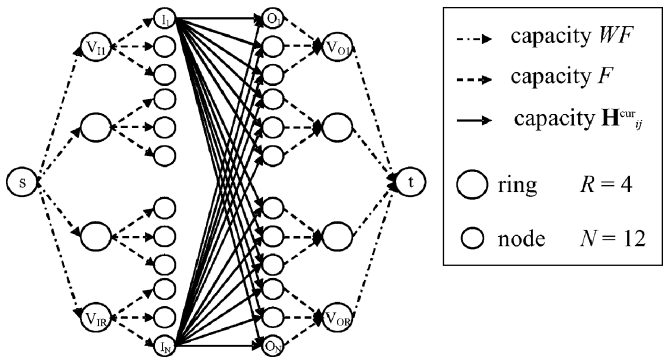


Fig. 6. Graph associated with the DAVID metro network.

which collects traffic from the n_i nodes it contains, and demultiplexer i represents ring i, which distributes traffic from the Hub to the n_i nodes it connects.

In this section, we present two optimum polynomial algorithms, one for the F-matching, and one for the TSA problem. Both make use of the Dinic algorithm [15] to calculate a maximum flow through the graph associated with the DAVID metro network shown in Fig. 6. The directed graph contains $2N+2R+2$ vertices:

- a source vertex s with R outgoing edges;
- R vertices (called $V_{I1}, \ldots, V_{IR}$) that represent source rings, each connected to as many vertices as the number of nodes on each ring;
- N vertices (called $I_1, \ldots, I_N$) that represent source nodes, each connected to any possible destination vertex by means of N edges;
- N destination vertices (called $O_1, \ldots, O_N$); R vertices (called $V_{O1}, \ldots, V_{OR}$) representing the ring to which destination nodes belong;
- a sink t with R incoming edges.

In the following sections, we will show how to use this graph for the solution of both the F-matching and the TSA problems.

C. F-Matching Algorithm

The outcome of the F-matching algorithm is an admissible request matrix $\mathbf{A}$ that must take into account the persistency of current HP connections ($\mathbf{H}^{\text{cur}}$ matrix) and the priority of HP traffic ($\mathbf{H}^{\text{cur}}$ and $\mathbf{\Delta}$ matrices) over BE traffic ($\mathbf{B}$ matrix). $\mathbf{A}$ is

obtained applying the Dinic algorithm three times. Initially, as represented in Fig. 6, edges connecting each node to its corresponding ring have capacity equal to F, and edges connecting rings to the source s or to the sink t have capacity WF; a capacity equal to $\mathbf{H}_{ij}^{\text{cur}}$ is assigned to the edge connecting node I_i to node O_j. The first run of the Dinic algorithm allocates current HP connections. Next, the capacities of edges connecting each node to its corresponding ring, and the capacities of edges connecting rings to s or to t are decreased to account for the connections just set up, and the capacity on edge (I_i, O_j) is set to $\mathbf{\Delta}_{ij}$. The Dinic algorithm is run again to allocate new HP connections and then, edge capacities are decreased accordingly. Finally, capacities on (I_i, O_j) are set equal to $\mathbf{B}_{ij}$ and BE traffic is allocated by running the Dinic algorithm for the third time. $\mathbf{A}_{ij}$ can be easily computed summing over the three runs of the Dinic algorithm the flow traversing edge (I_i, O_j).

The overall matching complexity can be shown to be $O(N^3)$, and does not depend on F[15].

D. Time Slot Assignment Algorithm

Once the F-matched matrix $\mathbf{A}$ is obtained, we need to allocate it in the frame. The TSA algorithm works with *full matrices* only, i.e., with matrices $\mathbf{A}^{\text{full}}$ for which

$$\sum_{i \in r^{-1}(k)} \sum_{j=1}^{N} \mathbf{A}_{ij}^{\text{full}} = WF, \quad \forall k \in \{1, 2, \ldots, R\}$$

$$\sum_{i=1}^{N} \sum_{j \in r^{-1}(k)} \mathbf{A}_{ij}^{\text{full}} = WF, \quad \forall k \in \{1, 2, \ldots, R\}$$

where $r^{-1}(k)$ is the set of the nodes that are on ring k. This also implies

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \mathbf{A}_{ij}^{\text{full}} = WFR.$$

If $\mathbf{A}$ is not a full matrix, we add dummy traffic to obtain a full matrix, named $\mathbf{A}^{\text{full}}$, to ensure that the overall schedule comprises exactly F permutations. The aim of the algorithm is to obtain a sequence of F binary switching matrices. Each switching matrix will have exactly WR nonnull elements (W for each ring), and will cover all *critical nodes* (i.e., nodes that need to transmit or to receive in each time slot in the frame for fulfilling their traffic requirements).

The problem of obtaining each of the F switching matrices can be mapped into a problem of maximum flow with lower bounds [16].

To apply the algorithm, capacities are modified and lower bound constraints are added as follows.

- Edges (s, V_{Ii}) have capacity W and lower bound W; the same holds for edges (V_{Oi}, t);
- Edges (V_{Ii}, I_j) and edges (O_j, V_{Oi}) have capacity 1 and lower bound 1 if node j is critical, zero otherwise;
- Node-to-node edges (I_i, O_j) have capacity $\mathbf{A}_{ij}^{\text{full}}$ and lower bound zero.

We apply the algorithm once per slot; the flows obtained on node-to-node edges give the first switching matrix. We subtract this matrix from $\mathbf{A}^{\text{full}}$ and repeat the process F times for the F time slots in the frame. Starting from the first switching matrix, wherever we find a non-null element, we assign it to HP traffic if there was an HP request. Otherwise, we assign it to BE traffic also if there was a BE request. If there were no requests both for BE and HP traffic, it means that we are considering the added dummy traffic, and we may drop it.

The overall TSA complexity can be shown to be $O(N^3F)$ [15].

V. Heuristic Scheduling for the FD Architecture

The heuristic approach is an incremental algorithm, which consists of three steps.

1) At first, all the slots that were allocated in the previous frame to BE traffic, as well as those corresponding to ended HP connections are released, so that only persistent HP connections remain allocated. The following two steps allocate new HP and BE traffic.
2) New HP requests (contained in $\mathbf{\Delta}$) are allocated first. The frame is scanned and HP requests are served in a round-robin fashion checking that all constraints are satisfied. This scheduling step ends when either all empty slots have been considered or all HP connection requests have been satisfied.
3) Finally, the remaining slots are used to allocate BE traffic contained in the *ring-to-ring* request matrix $\mathbf{B}^*$. Element $\mathbf{B}_{ij}^*$ represents the aggregate amount of traffic that nodes on ring i must transmit to nodes on ring j. $\mathbf{B}^*$ is scheduled independently of HP traffic through an iterated critical maximum size matching **[4]**. The set of ring-to-ring permutations obtained must be fit in the unused part of the frame by selecting the permutations that allocate the maximum number of slots. Note that the allocation for BE traffic is recomputed every time; hence, neither incremental allocation is implemented, nor persistency is enforced for BE traffic.

The complexity of this algorithm can be shown to be $O(N^2F)$ [15].

VI. Heuristic Scheduling for the FS Architecture

The complexity of the optimum scheduling algorithm for the FS architecture may be very high, because the Hub has to account for node positions along the rings and this makes contention resolution harder. We again resort to a heuristic algorithm, although it only permits to achieve suboptimal performance.

We propose a greedy algorithm similar to that for the FD architecture, which accounts for the additional constraints introduced by the FS configuration. We assume that a $WR \times F$ matrix $\mathbf{P}$ of is available at the Hub. If a slot is reserved for an HP connection or a BE packet, the source and destination node's addresses are stored in $\mathbf{P}$, as shown in Fig. 7. The aim of the heuristic algorithm is to fill in $\mathbf{P}$ maximizing the number of HP requests allocated, and then to use the remaining bandwidth to serve as much BE traffic as possible.

This approach assigns resources to HP and BE requests on a first-fit fashion enforcing the persistency of currently established HP connections. The algorithm is incremental: $\mathbf{\Delta}$ and $\mathbf{B}$

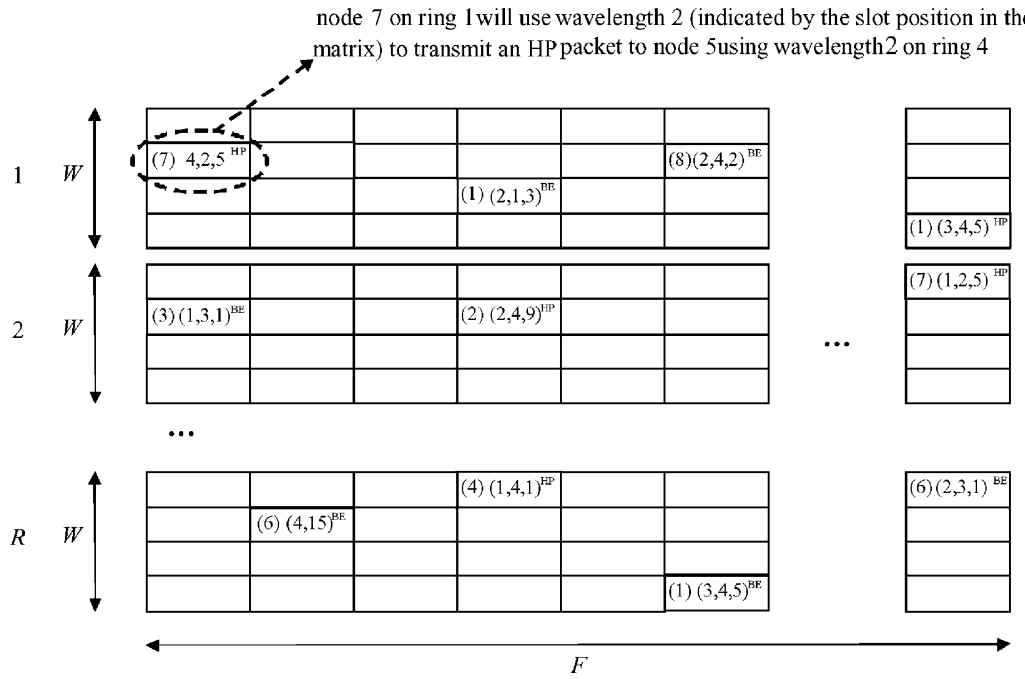


Fig. 7. Example of matrix $\mathbf{P}$ at the Hub.

only contain new HP and BE requests, respectively, while $\mathbf{P}$ stores the current slot allocation map. The scheduler performs the following steps.

1) Matrix $\mathbf{P}$ is scanned releasing all slots assigned to BE packets and all slot reserved for ended HP connections.
2) All HP requests that open a new connection or increase the already allocated bandwidth are scheduled. Matrix $\mathbf{P}$ is scanned in a round robin way, trying to allocate all the required slots and satisfy all constraints. This scheduling step ends when either all requests have been satisfied or all slots in the frame have been considered.
3) BE requests are scheduled using wavelength-to-wavelength permutations. Note that this step is more complex than the scheduling of BE traffic based on ring-to ring permutations performed by the heuristics in Section V.

The algorithm complexity is $O(N^2FW)$. Notice, however, that the algorithm complexity may be significantly reduced using the time-decoupling approach mentioned in Section III. See [17] for more details.

VII. Performance Evaluation

A. Simulation Scenario

To evaluate the performance of the algorithms presented in the previous sections, we study by simulation a network configuration comprising $R = 4$ rings and $n = 16$ nodes per ring ($N = 64$ nodes in total), each node sharing $W = 4$ wavelengths. The latter means that for the FS architecture each ring conveys five wavelengths (four for data and one for control), whereas, for the FD architecture, the wavelengths on each ring are nine (four for transmission traffic, four for reception traffic, and one for control). We set the slot size to 1 μs, the ring round-trip time to $\mathrm{RTT} = 512$ μs (512 slots on the ring), and the frame duration to $F = 10\,240$ slots (20 RTTs). Queues at nodes are considered infinite.

We assess network performance using four traffic patterns named: *uniform*, *diagonal*, *power-of-ten*, and *very unbalanced*. Traffic patterns are characterized by an $R \times R$ ring-to-ring matrix $\mathbf{K}$, whose generic element $\mathbf{K}_{ij}$ is a real number ranging between 0 and 1 representing the percentage of traffic generated on ring i toward ring j with respect to the total network load ρ. The four ring-to-ring matrices are as follows:

$$\text{Uniform}\quad \mathbf{K}_{\mathrm{u}}=\frac{1}{4}\begin{pmatrix}1&1&1&1\\1&1&1&1\\1&1&1&1\\1&1&1&1\end{pmatrix} \qquad \text{Power-of-ten}\quad \mathbf{K}_{\mathrm{p}}=\frac{1}{1111}\begin{pmatrix}1&10&10^2&10^3\\10^3&1&10&10^2\\10^2&10^3&1&10\\10&10^2&10^3&1\end{pmatrix}$$

$$\text{Diagonal}\quad \mathbf{K}_{\mathrm{d}}=\frac{1}{10}\begin{pmatrix}7&1&1&1\\1&7&1&1\\1&1&7&1\\1&1&1&7\end{pmatrix} \qquad \text{Very unbalanced}\quad \mathbf{K}_{\mathrm{v}}=\begin{pmatrix}\frac{1}{2}&0&0&0\\\frac{1}{2}&\frac{1}{10}&\frac{1}{3}&\frac{1}{15}\\0&0&\frac{1}{3}&0\\0&0&\frac{1}{3}&0\end{pmatrix}.$$

Matrix $\mathbf{K}_{\mathrm{u}}$ describes the uniform traffic pattern, whereas matrices $\mathbf{K}_{\mathrm{d}}$, $\mathbf{K}_{\mathrm{p}}$ and $\mathbf{K}_{\mathrm{v}}$ are used, respectively, to generate the diagonal, the power-of-ten and the very unbalanced traffic patterns.

In Figs. 8–10, we plot the normalized throughput (ratio between used and available slots) for each destination ring reachable from ring 1 under uniform, diagonal, and power-of-ten traffic patterns, respectively.

In all scenarios, the traffic generated by a node toward a given ring is uniformly distributed among the nodes belonging to that ring. BE packets are generated with a geometrically distributed interarrival time: all packets have the same size and fit in one slot. Instead, HP traffic is connection-oriented; each node generates connection requests occupying one slot per frame. Both the

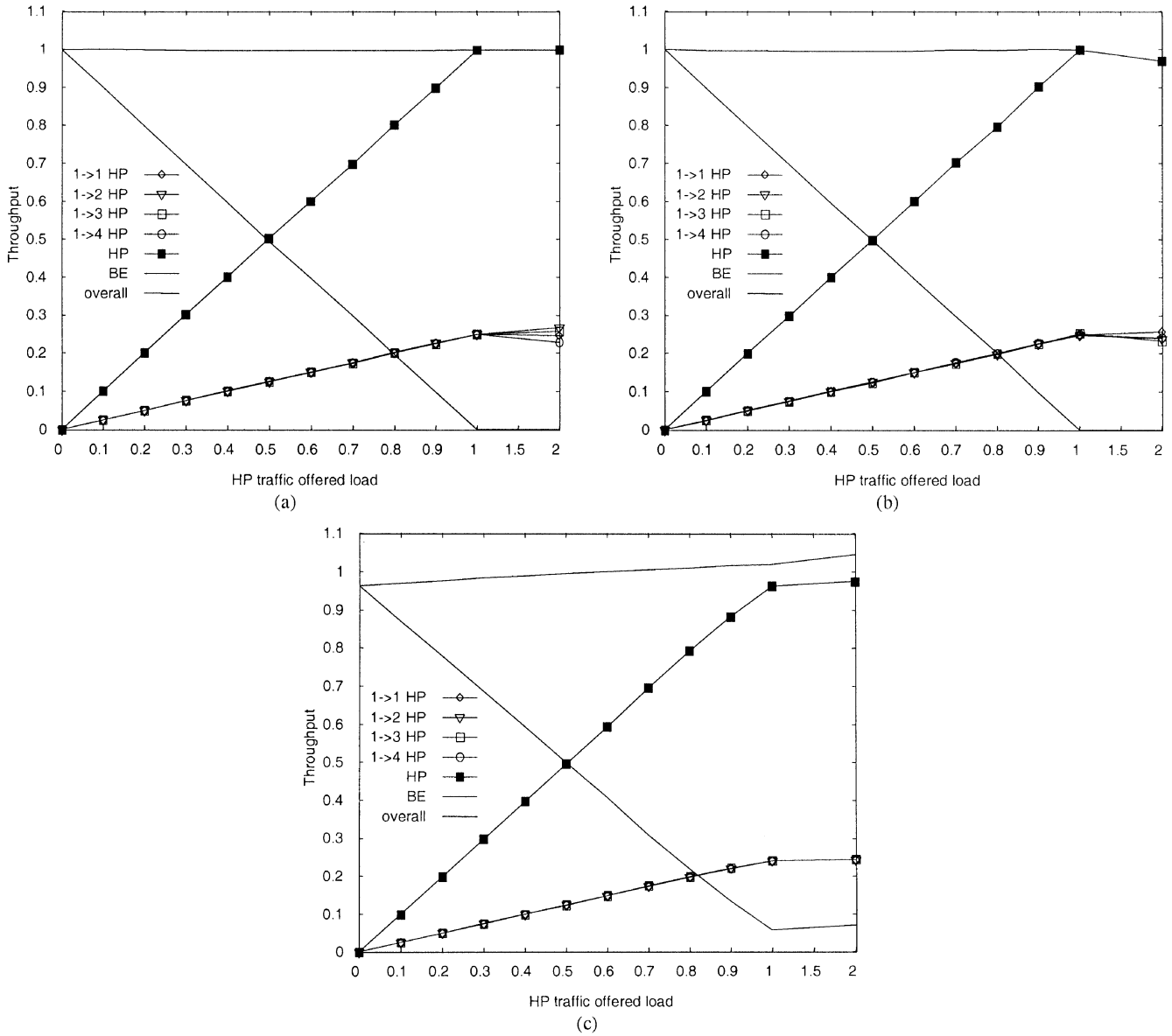


Fig. 8. Throughput as a function of HP traffic relative load under the uniform traffic pattern. Comparison of the optimal (a) and the heuristic solution for the FD configuration (b), with the heuristic solution for the FS architecture (c).

connection interarrival time and the connection duration are geometrically distributed. The mean value of the interarrival times for BE and HP traffic is selected accordingly to generate the required offered load ρ.

B. Simulation Results

In the following figures, we analyze the network performance when both traffic classes are present. All the points of the plots are steady-state values obtained from statistically significant measures.

All figures consist of three plots showing the performance of the optimum algorithm for the FD configuration and the heuristics for both the FD and the FS architecture.

All plots show the throughput as a function of the amount of HP traffic present in the network, when the total BE offered load is exactly 1. In other words, when the HP traffic load on the horizontal axis of the figures is 0.2, the total network load is 1.2. Note, however, that both BE and HP traffic are distributed among different rings according to the chosen ring-to-ring matrix $\mathbf{K}_x$.

The plots in Figs. 8–10 show the throughput for each destination ring on source ring 1 for HP traffic (white markers), the total HP throughput (black square markers), the total BE throughput (dashed line without markers), and the total throughput on ring 1 (solid line without markers). Although we plot the throughput for a single ring, the same behavior holds for all the other rings due to traffic symmetries.

Fig. 8 compares the three solutions under uniform traffic. Fig. 8(a) shows that HP throughput increases with the offered load until it reaches the value of 1 in overload. The overall throughput is constantly equal to 1, as it is always possible to fill with BE traffic the slots left free by HP connections. Fig. 8(b) presents the same behavior as Fig. 8(a) except when the offered

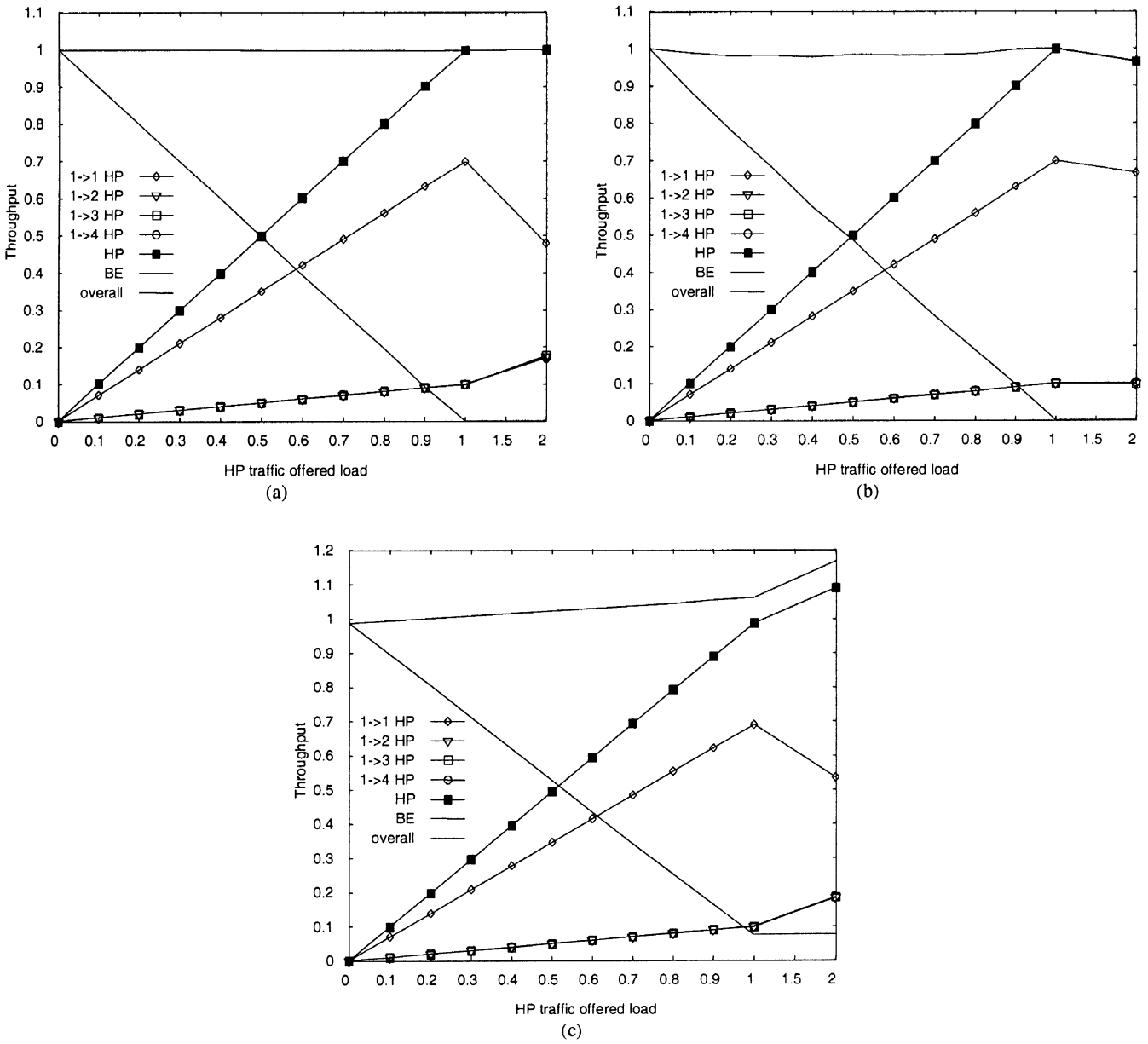


Fig. 9. Throughput as a function of HP traffic load under the diagonal traffic pattern. Comparison of the optimal (a) and the heuristic solution for the FD configuration (b), with the heuristic solution for the FS architecture (c).

load exceeds 1. In this case, the heuristics is not capable of maintaining the overall network throughput to 1.

In the FS configuration shown in Fig. 8(c), the network behaves differently. Indeed, in this case, we have half wavelengths and more contention probability due to shared transmission and reception channels; therefore, when only BE traffic is present in the network the overall network throughput is 0.97. As the HP traffic increases, the network throughput increases as well, reaching values even higher than 1. Note that in this case, the amount of BE traffic never drops to 0. Indeed, since we measure throughput as the ratio between the number of transmitted packets and the number of available slots, it may happen that a slot can be used more than once to transport different packets during one single round trip. For instance, a node can transmit a packet to a neighboring node on the same ring; the destination node can reuse the same slot to transmit another packet to a different node on the same ring, and so on. This is an implicit consequence of the wavelength reuse capability, and increases network throughput. Nevertheless, this gain can be exploited only for *intra-ring* traffic, i.e., when the transmitter and the receiver belong to the same ring. In any other case (i.e., for *inter-ring* traffic), packets must be switched at the Hub from their source ring to their destination ring and, therefore, slots containing such packets cannot be reused. For this reason, when traffic is mostly inter-ring, the gain obtained from wavelength reuse is lower: in Fig. 8(c), it reaches 1.05, and in Fig. 10(c), it is not even noticeable. In Fig. 11(c), instead, it becomes more evident, reaching 1.17, because the diagonal traffic pattern has a higher percentage of intra-ring traffic than the other scenarios.

In Fig. 9(a), obtained using the optimum algorithm for the FD architecture, the overall throughput is always 1, and HP throughput proportionally increases with the offered load. It is

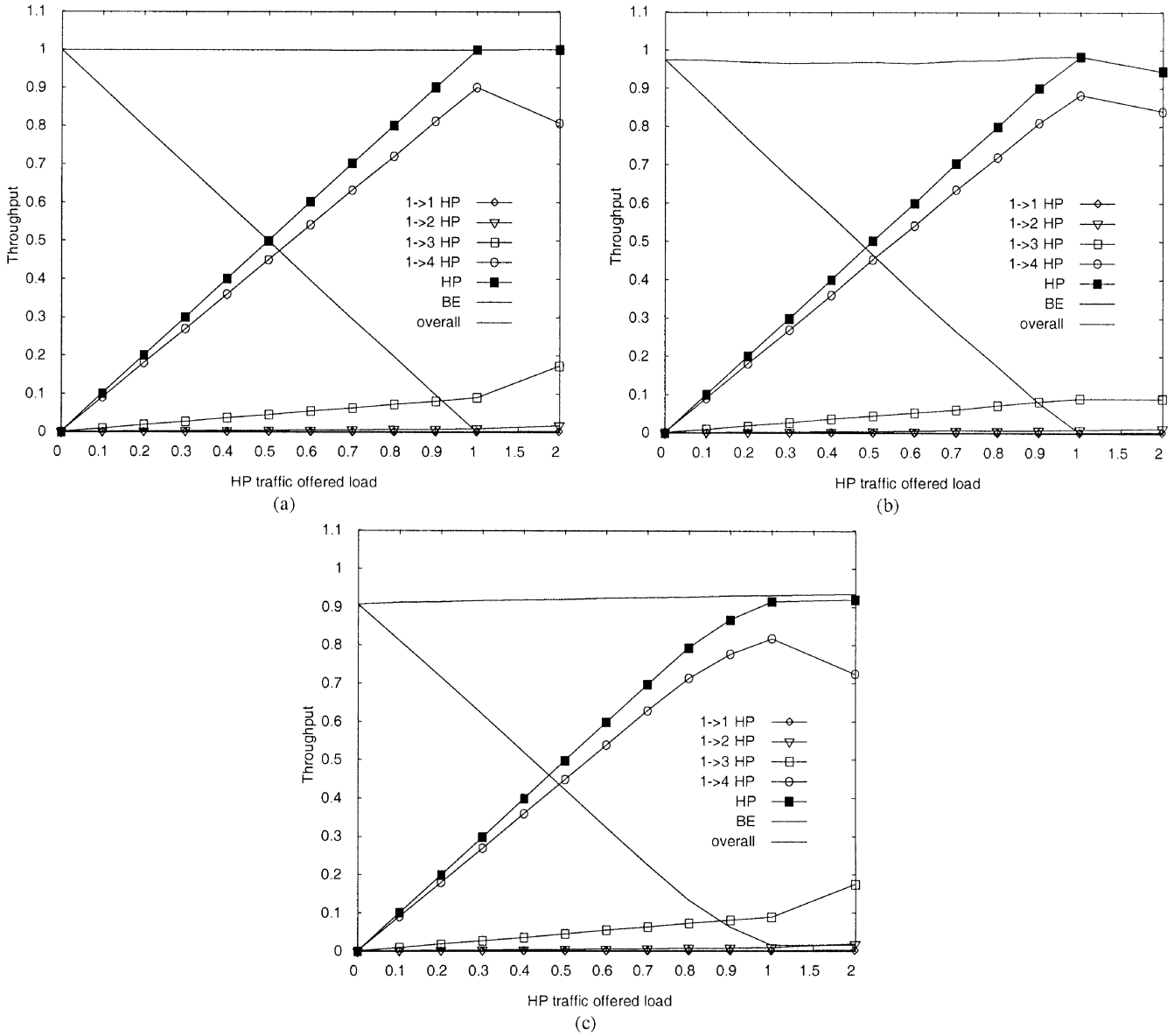


Fig. 10. Throughput as a function of HP traffic load under the power-of-ten traffic pattern. Comparison of the optimal (a) and the heuristic solution for the FD configuration (b), with the heuristic solution for the FS architecture (c)

interesting to note how intra-ring traffic, after reaching a value of about 0.7, starts to decrease and leaves resources to inter-ring traffic when the total HP traffic equals to the network capacity. This is due to the fact that the scheduler in overload tends to equalize the load on the rings, according to max–min throughput fairness. As before, the heuristic algorithm in Fig. 9(b) behaves similarly to the optimum one, except when the offered load exceeds 1. Moreover, the heuristics does not equalize rings' load.

In Fig. 10(a), we observe the optimum behavior of the algorithm with the power-of-ten traffic pattern. All HP connections are allocated optimally and total throughput remains equal to 1. This is not true for both heuristic algorithms. The heuristics for the FD configuration in Fig. 10(b) is not able to allocate all HP requests, reaching a throughput around 0.98 independent of the amount of HP traffic injected. The heuristics for the FS configuration in Fig. 10(c) performs even worse and the total network throughput does not rise above 0.9.

Finally, in Fig. 11, we analyze the case of the very unbalanced traffic pattern: Since it is not symmetric, in each subplot, we show the throughput of HP traffic on each ring, the total network throughput of HP and BE traffic, and the overall network throughput. The offered load is normalized with respect to ring 2, where the traffic is higher. This means that when HP traffic load in the horizontal axis of Fig. 11 is equal to 1, ring 1 is 50% loaded, ring 2 is 100% loaded, and rings 3 and 4 are 33.3% loaded; therefore, the total network load is about 0.54. Fig. 11(a) shows that while the allocation of HP traffic remains very close to the optimum, the very unbalanced traffic pattern causes total traffic allocation to be suboptimal. This is due to the nonoptimality of a double matching with two traffic classes. Indeed, when only one class is present in the network, the total throughput is always equal to 100%. Furthermore, heuristic algorithms in Fig. 11(b) and (c) are not able to allocate all traffic.

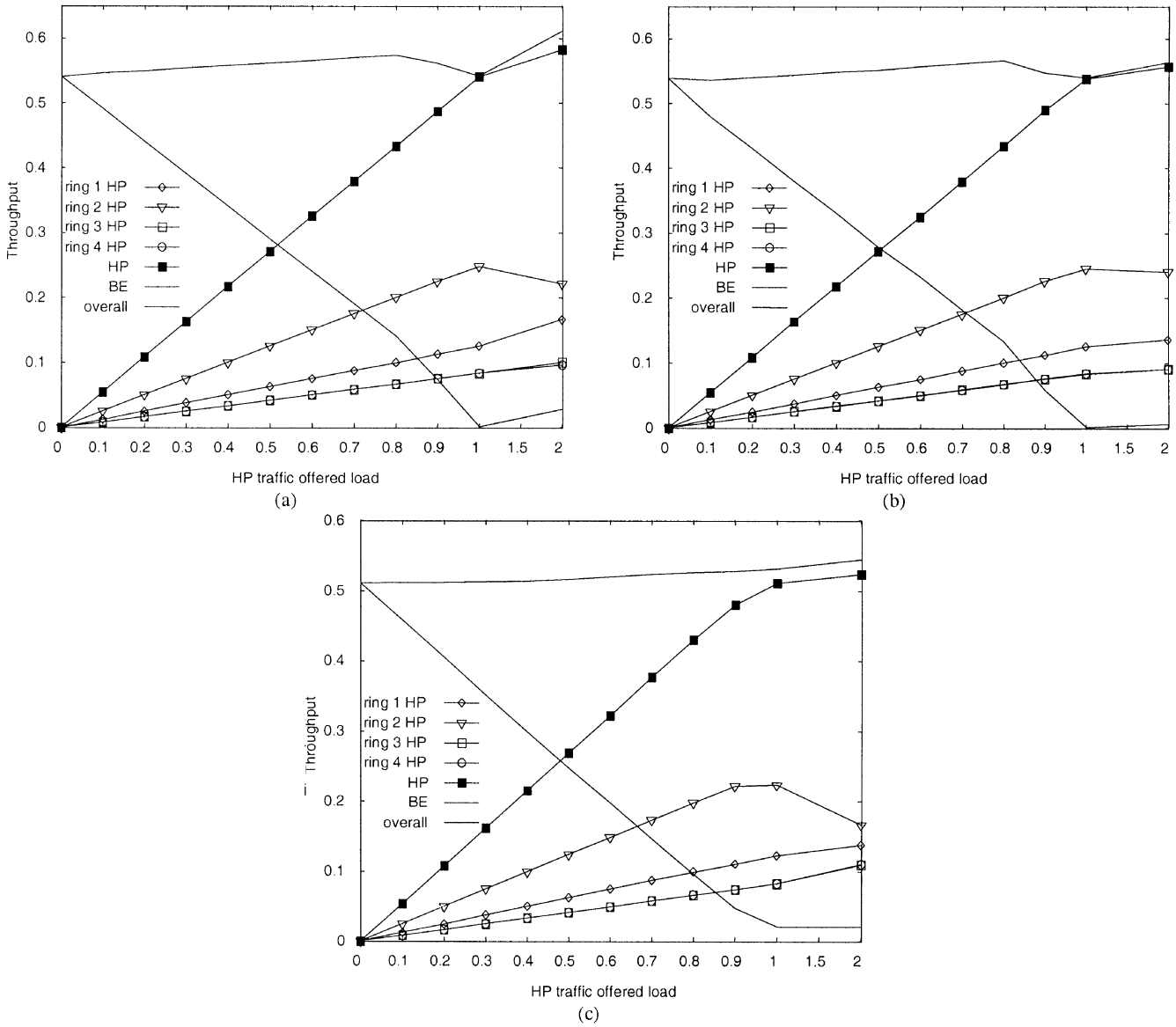


Fig. 11. Throughput as a function of HP traffic load under the very unbalanced traffic pattern. (a) Comparison of the optimal and (b) the heuristic solution for the FD configuration, and (c) with the heuristic solution for the FS architecture.

VIII. Conclusion

The DAVID European project studied advanced networking solutions based on optical packet switching. In this paper, we analyzed the problem of scheduling multiclass traffic in different configurations of the DAVID metro network.

The FD architecture requires less optical components and no active components along the data path, while the FS configuration requires fewer wavelengths. Nevertheless, in optical networks, a nonefficient fiber bandwidth usage may not have significant effects on network costs because first, bandwidth on network links is typically not a bottleneck in the optical domain, and second, network costs are mostly related to the number of electronic and optical interfaces at nodes, and not to the number of wavelengths used on the fiber.

Scheduling algorithms are run in a mostly centralized fashion at the Hub, but some access decisions may be distributed at network nodes, depending on the network configuration and on the desired level of performance guarantees.

Tradeoffs between optimality and complexity of the allocation schemes were studied by simulation in scenarios comprising two traffic classes. The complexity of the algorithms devised remains high, and, as a consequence, they are probably not suited for tracking instantaneous traffic variations. Our study, however, shows the flexibility of the DAVID network concept, and the effectiveness of the proposed strategies in accommodating very diverse traffic patterns. Indeed, heuristics perform very closely to the optimum algorithm.

Acknowledgment

The authors would like to thank their colleagues of the DAVID Project and the anonymous reviewers for their valuable suggestions.

REFERENCES

[1] L. Dittmana *et al.*, "The European IST project DAVID: A viable approach toward optical packet switching," *IEEE J. Select. Areas Commun.*, vol. 21, pp. 1026–1040, Sept. 2003.

[2] A. Bianco, G. Galante, E. Leonardi, and F. Neri, "Measurement based resource allocation for interconnected WDM rings," *Photon. Network Commun.*, vol. 5, no. 1, pp. 5–22, Jan. 2003.

[3] W. Stallings, *Local and Metropolitan Area Networks*, 6th ed. Englewood Cliffs, NJ: Prentice-Hall, 2000.

[4] B. Hajek and T. Weller, "Scheduling nonuniform traffic in a packet-switching system with small propagation delay," *IEEE/ACM Trans. Networking*, vol. 5, pp. 813–823, Dec. 1997.

[5] T. Inukai, "An efficient SS/TDMA time slot assignment algorithm," *IEEE Trans. Commun.*, vol. COM–27, pp. 1449–1455, Oct. 1979.

[6] C. S. Chang, W. J. Chen, and H. Y. Huang, "Birkhoff-von Neumann input buffered crossbar switches," in *Proc. IEEE INFOCOM*, vol. 3, Tel-Aviv, Israel, Mar. 2000, pp. 1614–1623.

[7] A. C. Kam and K.-Y. Siu, "Supporting bursty traffic with bandwidth guarantee in WDM distributed networks," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 2029–2040, Oct. 2000.

[8] A. Bianco, E. Leonardi, M. Mellia, and F. Neri, "Network controller design for SONATA – A large-scale all-optical passive network," *IEEE J. Select. Areas Commun.*, vol. 18, pp. 2017–2028, Oct. 2000.

[9] M. Karol, M. Hluchyj, and S. Morgan, "Input versus output queueing on a space division switch," *IEEE Trans. Commun.*, vol. COM–35, pp. 1347–1356, Dec. 1987.

[10] N. McKeown, A. Mekkittikul, V. Anantharam, and J. Walrand, "Achieving 100% throughput in an input-queued switch," *IEEE Trans. Commun.*, vol. 47, pp. 1260–1267, Aug. 1999.

[11] A. Stavdas, S. Sygletos, M. O'Mahoney, H. Lee, and C. Matrakidis, "IST-DAVID: Concept presentation and physical layer modeling of the metropolitan area network," *J. Lightwave Technol.*, vol. 21, pp. 372–383, Feb. 2003.

[12] H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. New York: Dover, 1998.

[13] A. Bianco, J. Finochietto, E. Leonardi, F. Marigliano, P. Mitton, F. Neri, and L. Quarello, "Multiclass resource allocation in interconnected WDM rings," in *Proc. Optical Network Design Modeling*, Budapest, Hungary, Feb. 2003, pp. 623–643.

[14] A. Varma and S. Chalasani, "An incremental algorithm for TDM switching assignments in satellite and terrestrial networks," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 364–377, Feb. 1992.

[15] R. E. Tarjan, *Data Structures and Network Algorithms*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1988.

[16] J. Evans and E. Minieka, *Optimization Algorithms for Networks and Graphs*, 2nd ed. New York: Marcel Dekker, 1992.

[17] D. Careglio, J. Solé-Pareta, and S. Spadaro, "Heuristics for providing guaranteed service in DAVID metro network," in *Proc. Eur. Conf. Optical Communications*, Rimini, Italy, Sept. 2003, Th.1.4.4, pp. 900–901.

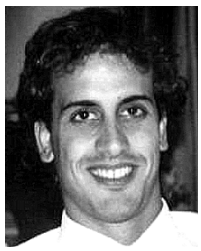
Andrea Bianco (M'90) was born in Turin, Italy, in 1962. He received the Dr.Ing. degree in electronics engineering and the Ph.D. degree in telecommunications engineering from Politecnico di Torino, Torino, Italy, in 1986 and 1993, respectively.

He is an Associate Professor in the Electronics Department, Politecnico di Torino. He has coauthored over 100 papers published in international journals and presented in leading international conferences in the area of telecommunication networks. His current research interests are in the fields of protocols for all-optical networks and switch architectures for high-speed networks.

Dr. Bianco was involved in several European (ACTS-SONATA, IST-DAVID), and Italian Projects (EURO, IPPO, WONDER) on optical networks and switch architectures. He has participated in the technical program committees of several conferences, including the IEEE INFOCOM 2000, the Quality-of-Service in Multiservice IP Networks 2001 (QoS-IP), the IFIP Working Conference on Optical Network Design and Modeling (ONDM) 2002, 2003, and 2004, and Networking 2002 and 2004. He was Technical Program Co-Chair of the High-Performance Switching and Routing (HPSR) 2003 Workshop.

Davide Careglio received the M.Sc. degree in electrical engineering from both the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 2000 and Politecnico di Torino, Torino, Italy, in 2001. He is currently working toward the Ph.D. degree at UPC.

He is an Assistant Professor in the Computer Architecture Department, UPC. He is a Member of the Advanced Broadband Communications Centre, UPC (http://www.ccaba.upc.es). He has recently been involved in the European Projects ACTS-SONATA, IST-LION, and IST-DAVID. His research interests are in the fields of all-optical networks with emphasis on MAC protocols, quality-of-service (QoS) provisioning, and traffic engineering.



Jorge M. Finochietto (S'99) was born in Buenos Aires, Argentina, in 1978. He received the degree in electronics engineering from the Universidad Nacional de Mar del Plata, Mar del Plata, Argentina, in 2000. Since 2002, he has been working toward the Ph.D. degree in the Dipartimento di Elettronica, Politecnico di Torino, Torino, Italy.

From 2000 to 2001, he was with the Engineering Group of Techtel, Buenos Aries, Argentina, a South American Network Operator, in the areas of routing performance, quality-of-service (QoS), and ATM. His research interests include the design of all-optical networks and switch architectures.

Giulio Galante (S'00–M'03) received the Dr.Ing. degree in electronics engineering and the Ph.D. degree in telecommunications engineering from Politecnico di Torino, Torino, Italy, in 1998 and 2003, respectively.

During Summer 2000, he was an Intern with Lucent Technologies, Bell Laboratories, Holmdel, NJ. From 2001 to 2002, he visited the research group of Prof. N. McKeown at Stanford University, Stanford, CA. Currently, he is with the Istituto Superiore Mario Boella, Torino. His research is mainly focused on the design of all-optical networks, the performance evaluation of wireless extensions to the TCP/IP protocol suite, and on open switching architectures.

Emilio Leonardi (S'94–M'99) was born in Cosenza, Italy, in 1967. He received the Dr.Ing degree in electronics engineering and the Ph.D. degree in telecommunications engineering from Politecnico di Torino, Torino, Italy, in 1991 and 1995, respectively.

He is currently an Assistant Professor in the Electronics Department, Politecnico di Torino. In 1995, he visited the Computer Science Department, University of California, Los Angeles (UCLA). He was with the High-Speed Networks Research Group, Bell Laboratories, Lucent Technologies, Holmdel, NJ (summer 1999), the Electrical Engineering Department, Stanford University, Stanford, CA (summer 2001), and the IP Group, Sprint, Advanced Technologies Laboratories, Burlingame CA (summer 2003). He has coauthored over 100 papers published in international journals and presented in leading international conferences, all of them in the area of telecommunication networks. His research interests are in the areas of performance evaluation of communication networks, all-optical networks, queueing theory, and packet switching.

Dr. Leonardi received the IEEE TCGN Best Paper Award for a paper presented at the IEEE GLOBECOM 2002 High-Speed Networks Symposium. He has participated in the program committees of several conferences including the IEEE INFOCOM, the IEEE GLOBECOM, and the IEEE ICC. He was Guest Editor of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS for two special issues that focused on high-speed switches and routers.

Fabio Neri (M'98) received the Dr.Ing. and Ph.D. degrees in electrical engineering from Politecnico di Torino, Torino, Italy, in 1981 and 1987, respectively.

He is a Full Professor in the Electronics Department, Politecnico di Torino. His teaching includes graduate-level courses on computer communication networks and on the performance evaluation of telecommunication systems. He leads a research group on optical networks at Politecnico di Torino. He has coauthored over 100 papers published in international journals and presented in leading international conferences. His research interests are in the fields of performance evaluation of communication networks, high-speed and all-optical networks, packet switching architectures, discrete event simulation, and queueing theory.

Dr. Neri was General Co-Chair of the 2001 IEEE Local and Metropolitan Area Networks (IEEE LANMAN) Workshop and General Chair of the 2002 IFIP Working Conference on Optical Network Design and Modeling (ONDM).

Josep Solé-Pareta (S'79–M'84) received the M.S. degree in telecommunication engineering and the Ph.D. degree in computer science from the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, in 1984 and 1991, respectively.

In 1984, he joined the Computer Architecture Department, UPC, where he has been an Associate Professor since 1992. He is Cofounder and Member of the Advanced Broadband Communications Centre, UPC (http://www.ccaba.upc.es). His current research interests are in broadband Internet and high-speed and optical networks. He is participating in NOBEL (IP-Project) and in e-Photon/One (Network of Excellence) of the European VI Framework Program.

Salvatore Spadaro received the M.Sc. degree in electrical engineering from both the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain, and Politecnico di Torino, Torino, Italy, in 2000. He is currently working toward the Ph.D. degree at UPC.

He is an Assistant Professor in the Signal Theory and Communications Department, UPC. He is a member of the Advanced Broadband Communications Centre, UPC (http://www.ccaba.upc.es). He has recently been involved in the European Projects ACTS-SONATA, IST-LION, and IST-DAVID. His research interests are in the fields of intelligent optical networks with emphasis on traffic engineering and resilience mechanisms.